

## CLAIMS

What is claimed is:

1. ~~A method for exchanging data messages between a first block having a first protocol for exchanging messages and a second block having a second protocol for exchanging message, the method comprising the steps of~~

receiving a first representation, representing the first protocol, said first representation using regular expressions;

receiving a second representation, representing the second protocol, said second representation using regular expressions;

generating a first finite automaton for said first representation;

generating a second finite automaton for said second representation;

generating a third representation, representing one or more permitted operations of said first and second finite automata; and

eliminating any non-determinisms in said third representation to generate an interface between said first and second protocols.

2. The method of claim 1, further comprising the step of:

automatically corresponding data from said first and second protocols.

3. The method of claim 2, further comprising the step of: automatically translating data between said first protocol and said second protocol, said data in said first protocol having a first sequence, said data in said second protocol having a second sequence that is different from said first sequence.

4. The method of claim 2, wherein said step of generating a first finite automaton includes the steps of:

identifying the initial state of the first protocol;  
identifying a first sequence of data according to the first protocol;  
constructing derivatives of said regular expressions; and  
eliminating equivalent expressions.

5. The method of claim 4, wherein said step of identifying a first sequence of data includes the steps of:

collecting data that is transferred during one or more transitions; and  
integrating said data with previous transitions.

6. The method of claim 5, further comprising the step of:

~~automatically translating data between said first protocol~~  
to said second protocol, said data in said first protocol having  
a first sequence, said data in said second protocol having a  
second sequence that is different from said first sequence.

7. The method of claim 1, further comprising the step of:  
automatically translating data between said first protocol  
to said second protocol, said data in said first protocol having  
a first sequence, said data in said second protocol having a  
second sequence that is different from said first sequence.

8. The method of claim 1, wherein the step of generating a third representation includes the steps of:

(a) selecting an interface state representing a first finite automaton state and a second finite automaton state;

(b) identifying all outgoing transitions in said selected state;

(c) determining a new state for each outgoing transition;

(d) repeating steps (a) - (c) for each interface state.

9. The method of claim 8, wherein the step of generating a third representation includes the steps of:

identifying said permitted operations as operations that do not result in a data inconsistency.

10. The method of claim 8, wherein said eliminating step includes the steps of:

identifying non-deterministic transitions for each interface state;

selecting a single outgoing transition for each interface state for each input value based upon priority parameters to generate a deterministic interface between the first and second protocols.

11. The method of claim 1, wherein said step of generating a first finite automaton includes the steps of:

identifying the initial state of the first protocol;

identifying a first sequence of data according to the first protocol;

constructing derivatives of said regular expressions; and eliminating equivalent expressions.

12. A computer based system for exchanging data messages between a first block having a first protocol for exchanging messages and a second block having a second protocol for exchanging message, the method comprising the steps of  
a storage device, for storing data, and sequences of operations;



second protocol having a second sequence that is different from said first sequence.

15. The system of claim 13, wherein said automata unit includes:

a first identifying unit for identifying the initial state of the first protocol;

a second identifying unit for identifying a first sequence of data according to the first protocol;

a derivative unit for constructing derivatives of said regular expressions; and

an eliminating unit for eliminating equivalent expressions.

16. The system of claim 15, wherein said second identifying unit includes:

a data collection unit for collecting data that is transferred as one or more transitions; and

a data analyzer for integrating said data with previous transitions.

17. The system of claim 12, further comprising:

a translation unit, for automatically translating data between said first protocol and said second protocol, said data in said first protocol having a first sequence, said data in said

second protocol having a second sequence that is different from said first sequence.

18. The system of claim 12, wherein the product unit includes:

a selection unit for selecting an interface state representing a first finite automaton state and a second finite automaton state;

an identifying unit for identifying all outgoing transitions in said selected state;

a state unit for determining a new state for each outgoing transition;

19. The system of claim 8, wherein the product unit includes:

a consistency unit for identifying said permitted operations as operations that do not result in a data inconsistency.

20. A computer program embodied in a tangible medium and capable of being read by a computer, for performing the method of claim 1.

*Adel  
A2*